

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-167756

(43)Date of publication of application : 13.06.2003

(51)Int.Cl.

G06F 11/22

G06F 11/28

(21)Application number : 2001-367496

(71)Applicant : FUJITSU TEN LTD

(22)Date of filing : 30.11.2001

(72)Inventor : HIGUCHI TAKASHI

IMADA SHOGO

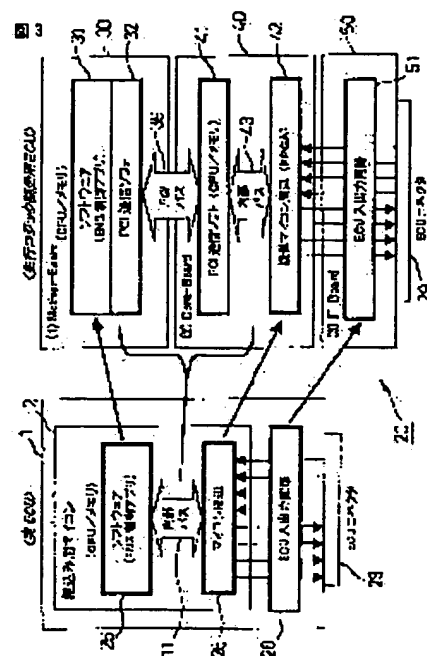
KASHIWABARA TOSHIHIRO

(54) LOGIC DEVELOPMENT DEVICE FOR MICROCOMPUTER

(57)Abstract:

PROBLEM TO BE SOLVED: To develop in a short period of time a built-in microcomputer capable of ensuring a CPU function necessary for advanced logic and ensuring the number of resources conforming to an advanced system.

SOLUTION: A logic development device for a microcomputer, which is a device for developing advanced logic for a built-in microcomputer used in a built-in relation to an electronic control unit, comprises a motherboard 30 in which a first CPU, a first memory, and a first interface for external communication are connected by a first internal bus, a core board 40 in which a second CPU, a second memory, a pseudo microcomputer peripheral device for implementing a pseudo microcomputer peripheral device via software, and a second interface for external communication are connected by a second internal bus, and a PCI bus 39 for connecting the motherboard and the core board. The development device replaces a built-in microcomputer to operate the advanced logic.



LEGAL STATUS

[Date of request for examination]

01.04.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

BEST AVAILABLE COPY

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2003-167756

(P2003-167756A)

(43) 公開日 平成15年6月13日 (2003.6.13)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 F 11/22	3 4 0	G 0 6 F 11/22	3 4 0 A 5 B 0 4 2
	3 6 0		3 6 0 B 5 B 0 4 8
11/28		11/28	L

審査請求 未請求 請求項の数23 O L (全 23 頁)

(21) 出願番号 特願2001-367496(P2001-367496)

(22) 出願日 平成13年11月30日 (2001.11.30)

(71) 出願人 000237592

富士通テン株式会社

兵庫県神戸市兵庫区御所通1丁目2番28号

(72) 発明者 樋口 崇

兵庫県神戸市兵庫区御所通1丁目2番28号

富士通テン株式会社内

(72) 発明者 今田 昭吾

兵庫県神戸市兵庫区御所通1丁目2番28号

富士通テン株式会社内

(74) 代理人 100077517

弁理士 石田 敬 (外4名)

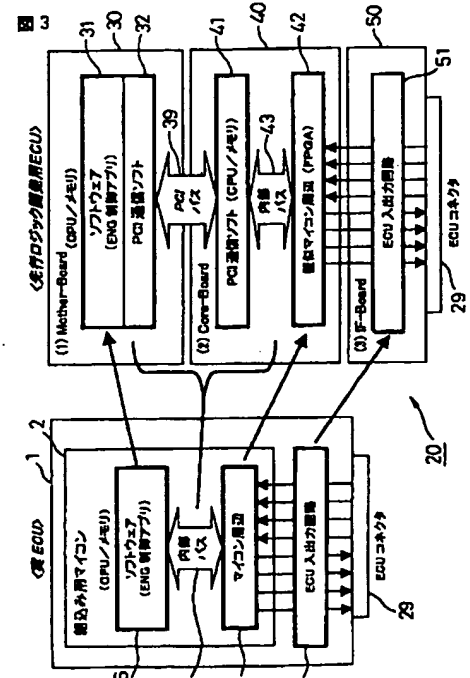
最終頁に続く

(54) 【発明の名称】 マイコンのロジック開発装置

(57) 【要約】

【課題】 先行ロジックに必要なCPU機能を確保でき、先行システムに合わせたリソース数を確保できる組み込み用マイコンを短時間で開発可能とする。

【解決手段】 電子制御ユニットに組み込まれて使用される組み込み用マイコン先行ロジックを開発する装置を、第1のCPU、第1のメモリ、外部との通信を行う第1のインタフェースとを第1の内部バスで接続したマザーボード30と、第2のCPU、第2のメモリ、マイコンの周辺装置を擬似的にソフトウェアで実現する疑似マイコン周辺装置、外部との通信を行う第2のインタフェースとを第2の内部バスで接続したコアボード40と、マザーボードとコアボードとを接続するPCIバス39とから構成し、この開発装置を組み込み用マイコンに置き換えて先行ロジックを動作させるようにしたマイコンのロジック開発装置である。



【特許請求の範囲】

【請求項1】 電子制御ユニットに組み込まれて使用される組込み用マイコンのロジックを開発する装置であって、

前記ロジック処理を行う第1の中央処理装置、前記ロジックのプログラムを含むデータを記憶する第1のメモリ、外部との通信を行う第1のインタフェース、及びこれらを接続する第1の内部バスとを少なくとも備えた第1のブロックと、

マイコンの周辺装置を擬似的にソフトウェアで実現する疑似マイコン周辺装置、外部との通信を行う第2のインタフェース、及びこれらを接続する第2の内部バスとを少なくとも備えた第2のブロックと、

前記第1と第2のブロックを接続するインタフェースバスとを備え、

これら第1と第2のブロック及びインタフェースバスによって、前記組込み用マイコンに置き換えて前記ロジックを動作させるようにしたことを特徴とするマイコンのロジック開発装置。

【請求項2】 前記第2のブロックに、前記第1の中央処理装置よりも処理能力が低く、且つ前記インタフェースバスによる通信を行うための第2の中央処理装置と、少なくとも前記通信に用いるデータを記憶する第2のメモリとが設けられていることを特徴とする請求項1に記載のマイコン用ロジック開発装置。

【請求項3】 更に、入出力回路が実装された第3のブロックを備え、この第3のブロックが前記第2のブロックに接続された状態で、前記電子制御ユニットに置き換えて使用できるようにしたことを特徴とする請求項1又は2に記載のマイコンのロジック開発装置。

【請求項4】 前記第1のブロックに、第1のタイマが設けられていることを特徴とする請求項1から3の何れか1項に記載のマイコンのロジック開発装置。

【請求項5】 前記第2のメモリが、前記インタフェースバスに接続する共有メモリと、前記第2の内部バスに接続する内部メモリから構成されることを特徴とする請求項1から4の何れか1項に記載のマイコンのロジック開発装置。

【請求項6】 前記第2のブロックにおいて、前記疑似マイコン周辺装置が追加可能に構成されていることを特徴とする請求項1から5の何れか1項に記載のマイコンのロジック開発装置。

【請求項7】 前記第2のブロックに、時間管理を行う第2のタイマが更に設けられていることを特徴とする請求項1から6の何れか1項に記載のマイコンのロジック開発装置。

【請求項8】 前記第1のメモリには、所定時間毎に実行される時間系処理アプリケーションと、所定の事象の発生毎に時間に関係なく実行される非時間系処理アプリケーションからなる制御用アプリケーションが備えられ

ており、

前記第1の中央処理装置には、少なくとも時間系割り込み処理と非時間系割り込み処理とを行う仮想割り込みコントローラ機能が備えられており、

前記第1のインタフェースには、少なくともデータ及び割り込み事象の情報を前記インタフェースバスを通じて送受信する通信ソフトが備えられており、

前記第2の中央処理装置は、前記第2のメモリと前記第2のインタフェースを使用して前記第1のインタフェースと、前記インタフェースバスを通じて割り込み事象とデータの授受に関する通信を行い、

前記疑似マイコン周辺装置には、入力機能と出力機能が備えられていることを特徴とする、請求項1に記載のマイコンのロジック開発装置。

【請求項9】 前記入力機能が、入力ポート、ラッチポート、A/D変換、及びキャプチャであり、前記出力機能が、出力ポート、パルス出力、コンペア、シリアルであることを特徴とする、請求項8に記載のマイコンのロジック開発装置。

【請求項10】 前記第2のメモリに、前記インタフェースバスに接続する共有メモリが設けられており、前記疑似マイコン周辺装置は、この共有メモリと前記インタフェースバスとを通じて、前記第1のメモリの時間系処理アプリケーションと非時間系処理アプリケーションとの間でデータのやり取りを行うことを特徴とする請求項8に記載のマイコンのロジック開発装置。

【請求項11】 前記共有メモリに通信同期カウンタが設けられており、この通信同期カウンタにより、前記疑似マイコン周辺装置と前記非時間系処理アプリケーションとの間のデータのやり取りの同期をとることを特徴とする請求項10に記載のマイコンのロジック開発装置。

【請求項12】 前記疑似マイコン周辺装置の処理起動タイミングを、所定のサンプリング周期で固定することなく任意とし、前記第1のブロックにおける前記非時間系処理アプリケーションの処理の終了をもって前記疑似マイコン周辺装置の次の処理起動タイミングとしたことを特徴とする請求項10又は11に記載のマイコンのロジック開発装置。

【請求項13】 前記疑似マイコン周辺装置の処理起動タイミングから、前記第1のブロックにおける前記非時間系処理アプリケーションの処理の終了までの時間が、予め定められた判定時間を越えた場合は、強制的に前記疑似マイコン周辺装置の処理の起動を行うことを特徴とする請求項12に記載のマイコンのロジック開発装置。

【請求項14】 前記疑似マイコン周辺装置で発生した割り込み事象を前記インタフェースバスを通じて前記第1のブロックに伝達して、前記非時間系処理アプリケーションの処理を行わせ、前記非時間系処理アプリケーションの処理の終了をもって前記疑似マイコン周辺装置の次の処理起動タイミングとしたことを特徴とする請求項

12又は13に記載のマイコンのロジック開発装置。

【請求項15】 前記非時間系処理アプリケーション側で、前記疑似マイコン周辺装置で発生する割り込み事象に優先順位を設定しておき、前記インタフェースバスを通じて前記第1のブロックに伝達された前記割り込み事象を、前記優先順位に従って前記非時間系処理アプリケーションが処理することを特徴とする請求項12から14の何れか1項に記載のマイコンのロジック開発装置。

【請求項16】 前記疑似マイコン周辺装置で発生する割り込み事象が何もない場合、前記インタフェースバスを通じて前記第1のブロックへの情報の伝達を省略するようにしたことを特徴とする請求項12から15の何れか1項に記載のマイコンのロジック開発装置。

【請求項17】 前記非時間系処理において、タイマ値取得要求時には、前記インタフェースバスにて取得したタイマ値を、前記第1のブロック内にある第1のタイマからのタイマ値で補正することを特徴とする請求項12から16の何れか1項に記載のマイコンのロジック開発装置。

【請求項18】 前記第2のブロックからの割り込み要求により、前記制御アプリケーションにおいて前記出力機能におけるコンペア機能を使用したパルス出力要求を前記第2のブロックに出力する場合に、前記制御アプリケーションからは、前記コンペア機能の出力端子に設けられた一般出力ポート機能を選択して即時出力を行うイミディエート出力と、前記コンペア機能の出力端子に設けられたコンペア出力機能を選択し、出力時刻、出力レベルを設定して出力予約を行うタイムド出力の、少なくとも一方を出力することを特徴とする請求項9に記載のマイコンのロジック開発装置。

【請求項19】 前記制御アプリケーションからの、前記イミディエート出力、或いは、タイムド出力に対して、これらの出力に何れの組み合わせに対しても、前記第2のブロックにおける前記疑似マイコン周辺装置が対応できるようにしたことを特徴とする請求項18に記載のマイコンのロジック開発装置。

【請求項20】 前記制御アプリケーションからの、前記イミディエート出力、或いは、タイムド出力が、前記インタフェースバスを通じて前記第2のブロックに伝達される際の、前記インタフェースバスを経由することによる、前記出力要求から実際の出力までの遅れ時間を補正するようにしたことを特徴とする請求項19に記載のマイコンのロジック開発装置。

【請求項21】 前記出力要求から実際の出力までの遅れ時間の補正が必要であるか、必要でないかを信号の種類に応じて判定し、補正が必要な種類の信号の場合のみ、前記補正を実行することを特徴とする請求項20に記載のマイコンのロジック開発装置。

【請求項22】 前記第1から第3のブロックがそれぞれ汎用のボードから構成されていることを特徴とする請

求項1から21の何れか1項に記載のマイコンのロジック開発装置。

【請求項23】 前記マイコンが、内燃機関制御用のものであることを特徴とする請求項1から22の何れか1項に記載のマイコンのロジック開発装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はマイコンのロジック開発装置に関し、特に、電子制御機器に組み込まれて使用されている組込み用マイコンにおけるロジックを開発するための装置に関する。

【0002】

【従来の技術】従来、電子制御機器の制御、例えば、ECU（電子制御ユニット）によって制御されるエンジンの制御は、排気ガス規制等の法規要求、及びマイクロコンピュータ（以後マイコンと記す）の性能向上に応じた性能向上を図るため、年々改良を加える必要があり、現状のECUの性能に先行して新しいロジックが開発されているのが現状である。このため、先行ロジックは、性能向上が見込まれる次期マイコンがターゲットとなることが多い。また、先行ロジックに必要とされる性能から性能の向上した次期マイコンの選定が行われる。

【0003】

【発明が解決しようとする課題】しかしながら、性能の向上した次期マイコンを組み込んだECUは実在しないため、前述の先行ロジックは、多くの場合は現状のマイコンをベースにして開発が行われる。

【0004】ところが、現状のマイコンを組み込んだECUを用いて先行ロジックの開発を行う場合には、以下のような問題点があった。

- (1) マイコンのCPUの処理能力が不足している。
- (2) マイコンのメモリの容量が足りない。
- (3) 周辺リソースが足りない。
- (4) 次期ECUを開発して製作する迄に時間がかかる。

【0005】そして、このような問題点の存在により、次期マイコンの開発が遅れ、この次期マイコンを組み込んだECUに制御される電子制御機器の新製品の開発に支障をきたしていた。

【0006】なお、現状の電子制御機器に組み込まれる組込み用マイコンは、電子制御機器のコストを低く抑えるために、現状のシステムに最適な仕様でCPU性能、及び周辺機能が選択されており、また、CPUやマイコン周辺リソースが1つのパッケージの中に組み込まれているため、それぞれの機能変更はマイコンを変更しない限り、不可能である。また、組込み用マイコンのロジックの開発を行うためには、CPU機能に対しては先行ロジックを処理するのに必要な処理性能が必要であり、マイコン周辺リソースには先行システムに合わせたリソースを確保する必要があった。更に、新たなマイコンの開

発を行う度に、マイコンに合わせて ECU を製作する必要があった。

【0007】そこで、本発明は、このような先行ロジックの開発に伴う課題を解消し、CPU 機能に対しては先行ロジックを処理するために必要な処理性能を確保することができ、マイコン周辺リソースに対しては、先行システムに合わせたリソースを確保することができて、先行ロジックを実現することができる組込み用マイコンを短時間で開発することが可能であると共に、繰り返し利用することが可能なマイコンのロジック開発装置を提供

【0008】

【課題を解決するための手段】前記目的を達成する本発明のマイコンのロジック開発装置は、電子制御ユニットに組み込まれて使用される組込み用マイコンのロジックを開発する装置であって、ロジック処理を行う第 1 の中央処理装置、ロジックのプログラムを含むデータを記憶する第 1 のメモリ、外部との通信を行う第 1 のインタフェース、及びこれらを接続する第 1 の内部バスとを少なくとも備えた第 1 のブロックと、マイコンの周辺装置を擬似的にソフトウェアで実現する疑似マイコン周辺装置、外部との通信を行う第 2 のインタフェース、及びこれらを接続する第 2 の内部バスとを少なくとも備えた第 2 のブロックと、第 1 と第 2 のブロックを接続するインタフェースバスとを備え、これら第 1 と第 2 のブロック及びインタフェースバスによって、組込み用マイコンに置き換えてロジックを動作させるようにしたことを特徴としている。

【0009】また、この構成を第 1 の形態として、第 2 のブロックに第 1 の中央処理装置よりも処理能力が低く、且つインタフェースバスによる通信を行うための第 2 の中央処理装置と、少なくとも通信に用いるデータを記憶する第 2 のメモリとが設けられた第 2 の形態と、入出力回路が実装された第 3 のブロックを備え、この第 3 のブロックが第 2 のブロックに接続された状態で、電子制御ユニットに置き換えて使用できる第 3 の形態が可能である。

【0010】更に、以下のような形態が可能である。

(1) 第 1 のブロックに、第 1 のタイマが設けられている第 4 の形態。

(2) 第 2 のメモリが、インタフェースバスに接続する共有メモリと、第 2 の内部バスに接続する内部メモリから構成される第 5 の形態。

(3) 第 2 のブロックにおいて、疑似マイコン周辺装置が追加可能に構成されている第 6 の形態。

(4) 第 2 のブロックに、時間管理を行う第 2 のタイマが更に設けられている第 7 の形態。

【0011】そして、第 1 の形態において、以下の構成が可能である。

【0012】第 1 のメモリには、所定時間毎に実行され

る時間系処理アプリケーションと、所定の事象の発生毎に時間に関係なく実行される非時間系処理アプリケーションとからなる制御アプリケーションが備えられており、第 1 の中央処理装置には、少なくとも時間系割り込み処理と非時間系割り込み処理とを行う仮想割り込みコントローラ機能が備えられており、第 1 のインタフェースには、少なくともデータ及び割り込み事象の情報を前記インタフェースバスを通じて送受信する通信ソフトが備えられており、第 2 の中央処理装置は、第 2 のメモリと第 2 のインタフェースを使用して第 1 のインタフェースとインタフェースバスを通じて割り込み事象とデータの授受に関する通信を行い、疑似マイコン周辺装置には入力機能と出力機能が供えられていることを特徴とする第 8 の形態。

【0013】第 8 の形態の入力機能が、入力ポート、ラッチポート、A/D 変換、及びキャプチャであり、出力機能が、出力ポート、パルス出力、コンペア、シリアルである第 9 の形態。

【0014】第 8 の形態において、第 2 のメモリに、インタフェースバスに接続する共有メモリが設けられており、疑似マイコン周辺装置は、この共有メモリとインタフェースバスとを通じて、第 1 のメモリの時間系処理アプリケーションと非時間系処理アプリケーションとの間でデータのやり取りを行う第 10 の形態。

【0015】第 10 の形態において、共有メモリに通信同期カウンタが設けられており、この通信同期カウンタにより、疑似マイコン周辺装置と非時間系処理アプリケーションとの間のデータのやり取りの同期をとる第 11 の形態。

【0016】第 10 又は第 11 の形態において、疑似マイコン周辺装置の処理起動タイミングを、所定のサンプリング周期で固定することなく任意とし、第 1 のブロックにおける非時間系処理アプリケーションの処理の終了をもって疑似マイコン周辺装置の次の処理起動タイミングとした第 12 の形態。

【0017】第 12 の形態において、疑似マイコン周辺装置の処理起動タイミングから、第 1 のブロックにおける非時間系処理アプリケーションの処理の終了までの時間が、予め定められた判定時間を越えた場合は、強制的に疑似マイコン周辺装置の処理の起動を行う第 13 の形態。

【0018】第 12 又は第 13 の形態において、疑似マイコン周辺装置で発生した割り込み事象をインタフェースバスを通じて第 1 のブロックに伝達して、非時間系処理アプリケーションの処理を行わせ、非時間系処理アプリケーションの処理の終了をもって疑似マイコン周辺装置の次の処理起動タイミングとした第 14 の形態。

【0019】第 12 から第 14 の何れかの形態において、非時間系処理アプリケーション側で、疑似マイコン周辺装置で発生する割り込み事象に優先順位を設定して

おき、インタフェースバスを通じて第1のブロックに伝達された割り込み事象を、優先順位に従って非時間系処理アプリケーションが処理する第15の形態。

【0020】第12から第15の何れかの形態において、疑似マイコン周辺装置で発生する割り込み事象が何もない場合、インタフェースバスを通じた第1のブロックへの情報の伝達を省略するようにした第16の形態。

【0021】第12から第16の何れかの形態において、非時間系処理において、タイマ値取得要求時には、インタフェースバスにて取得したタイマ値を、第1のブロック内にある第1のタイマからのタイマ値で補正する第17の形態。

【0022】第9の形態において、第2のブロックからの割り込み要求により、制御アプリケーションにおいて出力機能におけるコンペア機能を使用したパルス出力要求を第2のブロックに出力する場合に、制御アプリケーションからは、コンペア機能の出力端子に設けられた一般出力ポート機能を選択して即時出力を行うイミディエート出力と、コンペア機能の出力端子に設けられたコンペア出力機能を選択し、出力時刻、出力レベルを設定して出力予約を行うタイムド出力の、少なくとも一方を出力する第18の形態。

【0023】第18の形態において、制御アプリケーションからの、イミディエート出力、或いは、タイムド出力に対して、これらの出力に何れの組み合わせに対しても、第2のブロックにおける疑似マイコン周辺装置が対応できるようにした第19の形態。

【0024】第19の形態において、制御アプリケーションからの、イミディエート出力、或いは、タイムド出力が、インタフェースバスを通じて第2のブロックに伝達される際の、インタフェースバスを経由することによる、出力要求から実際の出力までの遅れ時間を補正するようにした第20の形態。

【0025】第20の形態において、出力要求から実際の出力までの遅れ時間の補正が必要であるか、必要でないかを信号の種類に応じて判定し、補正が必要な種類の信号の場合のみ、補正を実行する第21の形態。

【0026】なお、以上の全ての形態において、第1から第3のブロックをそれぞれ汎用のボードから構成することができる。また、本発明のマイコンのロジック開発装置は、内燃機関の制御用のマイコンに有効に適用できる。

【0027】以上のように構成された本発明のマイコンのロジック開発装置によれば、マイコンのロジックの開発に伴う課題が解消され、CPU機能に対しては新規なロジックや次期ロジックを処理するために必要な処理性能を確保することができ、マイコン周辺リソースに対しては、新規システムや次期システムに合わせたリソースを確保することができて、新規ロジックや次期ロジックを実現することが可能な組込み用マイコンを短時間で開

発することが可能となる。また、本発明のマイコンのロジック開発装置は、ロジックの開発に際して繰り返し利用することが可能であるので、開発コストを低減することができる。

【0028】

【発明の実施の形態】以下添付図面を用いて本発明の実施形態を具体的な実施例に基づいて詳細に説明するが、以下の実施例では、本発明を適用する電子制御機器として、電子制御式内燃機関（内燃機関は以下エンジンと記す）を説明する。

【0029】図1は従来の電子制御式エンジンの制御システムにおけるECU（電子制御ユニット）1の構成を示すシステム構成図である。電子制御式エンジンでは、エンジン回転数信号や車速信号等のパルス入力、水温センサや吸気温センサ等からのアナログ入力、及びスタータスイッチ、電気負荷スイッチ、シフト位置スイッチやエアコン信号等のデジタル入力がECU1に入力される。ECU1は、これらの入力信号を処理する組込み用マイコン2と、組込み用マイコン2で処理された信号を増幅して出力するECU入出力回路であるドライバ16とを備えて構成される。このECU1から出力されるのは、シフト制御ソレノイドやVVT（可変バルブタイミング）ソレノイドへのアナログ出力、点火信号や燃料の噴射信号等のパルス出力、ISC（アイドル速度制御）用のパルス出力、及び、チェックエンジンランプ、メインリレーやエヤコンカット信号等のデジタル信号等である。

【0030】組込み用マイコン2は、演算処理を行うメモリ9とCPU10、及び入出力（I/O）制御を行う周辺リソースが、1つのパッケージに収納されたものである。周辺リソースには、入力系のリソースと出力系のリソースとがある。図1には、入力系のリソースとして、デジタル信号を扱う入力ポート3とラッチポート4、アナログ入力を扱うA/Dコンバータ5、及びパルス入力を扱うキャプチャ6が示してあり、出力系のリソースとしては、デジタル出力を出力する出力ポート12、パルス出力を出力するPWM（パルス幅変調器）13とコンペア14、及びアナログ出力を入出力するシリアル15が示してある。これらの周辺リソースは内部バス11によってメモリ9及びCPU10に相互に接続されている。また、組込み用マイコン2の内部には、これらの周辺リソースに加えて、内部タイマ7や割り込みコントローラ8が設けられている。

【0031】電子制御式エンジンの制御システムでは、車両の運転状態を表す各センサやスイッチ類からの信号がECU1に取り込まれる。ECU1の入力回路では入力信号が信号処理され、組込み用マイコン2に入力される。入力された信号は前述の入力系の周辺リソースでCPU値に変換され、演算部であるメモリ9とCPU10では入力信号から車両状態が検出され、車両状態に応じ

た出力要求信号が作られる。この出力要求信号は前述の出力系の周辺リソースで出力信号に変換され、組込み用マイコン2から出力される。ECU1の入出力回路であるドライバ16はこの出力信号に従って車両に装備された各アクチュエータを駆動し、この出力制御の結果が破線で示すように、車両からの入力信号に反映される。

【0032】図2は、図1で説明したECU1の自動車(車両)18への搭載位置を示すものである。ECU1は車両18のエンジン19がマウントされるエンジンルームに搭載される。また、図2には、本発明のマイコンのロジック開発装置20が示してある。本発明のマイコンのロジック開発装置20は、この図に示すように、車両18に搭載されたECU1に接続するコネクタを外し、このコネクタに接続コード20Aによって直接接続して使用することができる。21はマイコンのロジック開発装置20の状態をモニタするための表示器、22はマイコンのロジック開発装置20の設定を変更するための入力装置であるキーボードである。

【0033】なお、本発明のマイコンのロジック開発装置20は、このように車両18に直接接続して使用することができる他に、パーソナルコンピュータ24の制御によって動作して、車両の色々な運転状況を擬似的に発生することができる車両の運転状況発生装置23に接続すれば、車両が無い状態でも、電子制御式エンジン用の組み込みマイコンのロジックを開発することができる。

【0034】図3は、図2のように接続して使用することができる、本発明のマイコンのロジック開発装置20のシステム構成を、従来の電子制御式エンジン(電子制御機器)用のECU1の構成と比較して示すものである。前述のように、ECU1には、組込み用マイコン2と、ドライバ16から構成されるECU入出力回路28があり、ECUコネクタ29で車両側の電子機器と接続されている。また、組込み用マイコン2の中には、メモリ9に格納されていてCPU10によって読み出されて使用されるソフトウェア(エンジン制御アプリケーション：図にはENG制御アプリと記載)25と、マイコン周辺リソース26とがあり、内部バス11で相互にデータのやり取りができるようになっている。

【0035】一方、図2で説明したように、このECU1に置き換えて使用する本発明のマイコンのロジック開発装置20は、この実施例では、マザーボード30、コアボード40、及びIFボード50の3つのボードから構成されている。マザーボード30とコアボード40がECU1の組込み用マイコン2に対応するものであり、IFボード50がECU1のECU入出力回路28に対応するものである。そして、マザーボード30とコアボード40とは、高速のバスインタフェースであるPCIバス39で接続されている。

【0036】マザーボード30には、後述するメモリに格納されていてCPUによって読み出されて使用される

ソフトウェア(ENG制御アプリ)31と、PCIバス39を通じて通信を行うためのPCI通信ソフトウェア(以後ソフトウェアはソフトと略記)32が設けられている。PCIバス39を用いたPCI通信処理は、疑似マイコン周辺リソース(以後周辺リソースは単に周辺と略記)42とやり取りするデータを、PCIバス39に載せる通信処理である。このマザーボード30には、次期のECUを開発するに当たっては、次期ECUの先行ロジックの開発に耐え得る演算性能、メモリ容量を備えさせることが重要である。

【0037】なお、現状のエンジン制御用のマイコンの性能は、CPUが66MHz、メモリが800Mバイト程度であるので、パソコン等に用いられている汎用のものを用いれば、十分すぎる性能であるといえ、長期間にわたって何度でも使用することが可能となる。

【0038】また、コアボード40には、CPUとメモリを含み、前述のPCIバス39と通信を行うためのPCI通信ソフト41と、組込み用マイコン2のマイコン周辺リソース26に対応する疑似マイコン周辺(FPGA:Field Programmable Gate Array)42とがあり、内部バス43で相互にデータのやり取りができるようになっている。IFボード50には、ECU1のECU入出力回路28に対応するECU入出力回路51と、ECUコネクタ29とが設けられている。ECU入出力回路51は、標準回路ブロック単位で独立させ、その組み合わせで構成し、入出力回路の変更に対して柔軟に対応できるようにする。

【0039】図4は、図3のマザーボード30とコアボード40の、ハードウェアの構成の一実施例を示すものである。マザーボード30には、図3で説明したソフトウェア(ENG制御アプリ)を記憶するための記憶容量が大きいメモリ31、汎用の高性能のCPU(例えば、動作周波数が850MHz)33、内部タイマ35、PCIバス39に接続するPCIバスインタフェース36、及びこれらを相互に接続する内部バス37がある。

【0040】コアボード40には、PCIバス39に接続するPCIバスインタフェース44、マザーボード30に搭載されたCPU33よりも低グレードのCPU45、マイコン周辺の機能と同等の機能を実現する疑似マイコン周辺(FPGA)42、内部バス43、PCIバスに接続する共有メモリ46、及び内部バス43に接続する内部メモリ47がある。また、コアボード40に搭載するCPU45は、PCI通信処理を行うことができる程度の処理能力(例えば、汎用32ビットCPUで動作周波数が16MHz程度)であれば良い。コアボード40の機能は、従来のECU1におけるENG制御アプリ(ソフト)25とマイコン周辺26とがやり取りするデータをPCIバス39を経由して受け、疑似マイコン周辺(FPGA)42へ受け渡すことである。

【0041】IFボード50に接続される各FPGA4

2はソフトで組むことができ、マイコン周辺の変更に柔軟に対応させることができる。即ち、チャンネル数を増やしたい場合や、新しい機能のリソースを追加したい等の場合に対応させることができる。

【0042】従来は、組込み用マイコン2のリソースが不足する場合は、図5(a)に示すように、組込み用マイコン2の前段部に外付けADC(A/Dコンバータ)17を追加していたが、本発明のマイコンのロジック開発装置20では、マザーボードのメモリ9の容量が大きく、また、CPU10の処理速度が速いので、図5

(a)のADコンバータ5と外付けADコンバータ17を合わせた、図5(b)に示すような、マイコンのリソースに必要な入力数を備えたADコンバータ5Aを容易に実現することができる。このため、本発明のマイコンのロジック開発装置20には、マイコンのリソースの不足を補うための拡張I/O(入出力装置)機能は不要となる。

【0043】図6は、本発明のマイコンのロジック開発装置20における、マザーボード30とコアボードのソフト構成を示す構成図である。電子制御機器が電子制御式エンジンの場合は、マザーボード30におけるENG制御アプリ31は、時間系割り込み処理と非時間系割り込み処理の2つの処理に分けられる。時間系割り込み処理は、外部状態に依存せず、一定の時間感覚で定期的に行う処理である。一方、非時間系割り込み処理は外部状態、即ち、エンジンの運転状態に依存して発生する処理である。この非時間系割り込み処理には、例えば、エンジンの回転信号、車速信号、燃料の噴射タイミング信号、点火タイミング信号等による割り込み処理がある。

【0044】ここで、このような外部状態により変化する事象をイベントと呼ぶこととすると、非時間系割り込み処理は、外部状態によりイベント情報を検出して、そのイベントに同期して行う処理とすることができる。図6には、このイベントの流れが破線で示され、データの流れが実線で示されている。

【0045】マザーボード30のENG制御アプリ31において、前述のような時間系割り込み処理と非時間系割り込み処理とを行うために、CPU33は仮想マイコン周辺として機能する。この仮想マイコン周辺には、時間系割り込みと非時間系割り込みを発生させる仮想割り込みコントローラ34と、仮想I/Oレジスタ38が設けられている。また、PCI通信ソフト32は、データとして、割り込みイベント情報と、I/Oレジスタデータとを扱う。35は時間系割り込み処理に使用される内部タイマである。仮想I/Oレジスタ38とPCIバス39におけるタイマ値については後述される。

【0046】一方、コアボード40側の疑似マイコン周辺42には、従来のマイコン周辺26と同様に、ポート、ラッチ、PWM、シリアル、コンペア、及びキャプチャの各機能が設けられている。また、PCI通信ソフト

ト41は、データとして、割り込みイベント情報と、I/Oレジスタデータとを扱う。更に、コアボード40には、イベントの流れを発生させるタイマ48が内蔵されている。また、PCIバス39には、割り込み指示用の1系統のバスが含まれる。

【0047】次に、図6に示すようにイベント情報とデータとが流れる本発明のマイコンのロジック開発装置20におけるPCI通信方法、通信同期方法、フェールセーフ方法、割り込み制御方法(割り込みイベントのセットとクリア方法、割り込み優先度管理方法)、PCI通信の処理負荷の軽減方法、タイマ処理方法、コンペア機能を使用したパルス出力処理方法、パルス出力要求方法、及びパルス出力補正方法について、具体的な実施例に基づいて詳細に説明する。

(1) PCI通信方法(共有メモリの使用)

図7は、本発明のマイコンのロジック開発装置20における、図4で説明した共有メモリ46を用いたコアボード40とマザーボード30との通信方法の一実施例を示すものである。また、図8は、図7の通信方法を実現するためにコアボード40に設置される共有メモリ46の構成及び機能を示すものである。

【0048】図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期(例えば、100μs)になると、ステップ701において出力情報のPCI通信ソフト41へのセットが行われ、共有メモリ46から出力情報が取得され、疑似マイコン周辺(FPGA)42に渡される。次のステップ702では、入力情報のセットが行われ、疑似マイコン周辺(FPGA)42から入力情報が取得され、共有メモリ46に格納される。そして、ステップ703ではコアボード40からマザーボード30への割り込み要求が行われてこのルーチンが終了する。

【0049】マザーボード30側では、このコアボード40からの割り込み要求により、非時間系処理が開始される。ステップ711では、PCI通信ソフト32によってPCI受信処理が行われ、PCIバス39経由で共有メモリ46から入力された情報が読み出され、ENG制御アプリ31に公開される。ステップ712では、ENG制御アプリ31において非時間系処理が行われる。そして、ステップ713においてはPCI送信処理が行われ、PCI通信ソフト32により、ENG制御アプリ31における非時間系処理の算出結果の出力情報が、PCIバス39経由で共有メモリ46に書き込まれてこのルーチンが終了する。

【0050】一方、マザーボード30側では、コアボード40からの割り込み要求とは関係なしに、ENG制御アプリ31において時間系処理が実行される。この時間系処理は、マザーボード30内のタイマ35により、例えば、1msの周期で実行される。この時間系処理では、ステップ721においてPCI通信ソフト32によ

ってPCI受信処理が行われ、PCIバス39経由で共有メモリ46から入力された情報が読み出され、ENG制御アプリ31に公開される。ステップ722では、ENG制御アプリ31において時間系処理が行われる。そして、ステップ723においてはPCI送信処理が行われ、PCI通信ソフト32により、ENG制御アプリ31における時間系処理の算出結果の出力情報が、PCIバス39経由で共有メモリ46に書き込まれてこのルーチンが終了する。

(2) PCI通信方法（通信同期方法1）

PCIバス39を経由してマザーボード30とコアボード40の間でデータのやり取りを行う場合は、共有メモリ46に対してマザーボード30から書き込みを行った値をコアボード40で読み出しを行う。また、その逆も行われる。この場合、書き込みタイミングと読み出しタイミングが重なった時、データの値の保証ができなくなる。そこで、本発明では、それぞれのタイミングが重ならないように、通信に同期をとってPCIバス39のアクセスを実行する。この通信同期の方法を図9を用いて説明する。

【0051】PCIバス39の通信に同期をとる場合は、共有メモリ46に通信同期カウンタを設け、マザーボード30の非時間系処理とコアボード40との間で処理の同期をとるようにする。通信同期カウンタは、マザーボード30側での処理が完了すると更新されるようにする。また、コアボード40側の処理では、通信同期カウンタが更新されていることを確認して処理を行う。この処理を図9を用いて説明する。

【0052】図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期（例えば、100μs）になると、ステップ901において通信同期カウンタの値（syncnt）が通信同期カウンタモニタの値（syncnt_m）に等しいか否かが判定される。等しくない場合はこのままこのルーチンを終了するが、等しい時はステップ902に進み、出力情報のセットが行われる。次のステップ903では入力情報のセットが行われ、続くステップ904においてマザーボード30への割り込み要求が行われる。そして、次のステップ905で通信同期カウンタモニタの値（syncnt_m）が1だけ更新されてこのルーチンが終了する。

【0053】マザーボード30側では、このコアボード40からの割り込み要求により、非時間系処理が開始される。ステップ911では、PCI通信ソフト32によってPCI受信処理が行われ、PCIバス39経由で共有メモリ46から入力された通信同期カウンタの値（syncnt）が、ステップ912でENG制御アプリ31に値（t_syn）として記憶される。ステップ913では、ENG制御アプリ31において非時間系処理が行われる。そして、ステップ914においてはPCI送信処理が行われ、ステップ915でステップ912で記憶された値（t

_syn）が1だけ更新されて通信同期カウンタの値（syncnt）として記憶される。ENG制御アプリ31における非時間系処理の結果と通信同期カウンタの値（syncnt）は、PCI通信ソフト32により、PCIバス39経由で共有メモリ46に書き込まれてこのルーチンが終了する。

【0054】一方、マザーボード30側では、コアボード40からの割り込み要求とは関係なしに、ENG制御アプリ31において時間系処理が実行される。この時間系処理は、マザーボード30内のタイマ35により、例えば、1msの周期で実行される。この時間系処理では、ステップ921においてPCI通信ソフト32によってPCI受信処理が行われる。このとき、非時間系処理で取得された最新の値が使用される。ステップ922では、ENG制御アプリ31において時間系処理が行われる。そして、ステップ923においてはPCI送信処理が行われ、データの更新のみ次の非時間系送信処理で送信されてこのルーチンが終了する。

【0055】図10は、図9で説明した通信同期方法の処理の進行状況を時間と共に示したタイミングチャートである。図10には、通信処理がコアボード40のI/Oサンプリング周期（100μs）で終了した場合の、通常時のタイミングチャートと、通信処理がコアボード40のI/Oサンプリング周期（100μs）後の、次の処理開始までに完了しなかった場合の非通常時のタイミングチャートとが示してある。

(3) PCI通信方法（通信同期方法2）

この実施例も、PCIバス39を経由したマザーボード30とコアボード40の間のデータのやり取りにおける書き込みタイミングと読み出しタイミングの同期をとるものである。（2）の方法との違いは、コアボード40の処理起動タイミングを、I/Oサンプリング周期で固定するのではなく、マザーボード30側の処理状況を確認して、処理が完了していれば、次の処理に移行するようにした点である。このため、コアボード40側のI/O処理タイミングは任意となる。この通信同期の方法を図11を用いて説明する。

【0056】コアボード40の電源がオンになると、ステップ1101において通信同期カウンタの値（syncnt）が通信同期カウンタモニタの値（syncnt_m）に等しいか否かが判定される。等しくない場合はこの値が等しくなるまで、ステップ1101の判定が繰り返される。ステップ1101の判定が等しい時はステップ1102に進み、出力情報のセットが行われる。次のステップ1103では入力情報のセットが行われ、続くステップ1104においてマザーボード30への割り込み要求が行われる。そして、次のステップ1105で通信同期カウンタモニタの値（syncnt_m）が1だけ更新されてこのルーチンが終了する。

【0057】マザーボード30側では、このコアボード

40からの割り込み要求により、非時間系処理が開始される。ステップ1111では、PCI通信ソフト32によってPCI受信処理が行われ、PCIバス39経由で共有メモリ46から入力された通信同期カウンタの値(syncnt)が、ステップ1112でENG制御アプリ31に値(t_syn)として記憶される。ステップ1113では、ENG制御アプリ31において非時間系処理が行われる。そして、ステップ1114においてはPCI送信処理が行われ、ステップ1115でステップ1112で記憶された値(t_syn)が1だけ更新されて通信同期カウンタの値(syncnt)として記憶される。ENG制御アプリ31における非時間系処理の結果と通信同期カウンタの値(syncnt)は、PCI通信ソフト32により、PCIバス39経由で共有メモリ46に書き込まれてこのルーチンが終了する。

【0058】一方、マザーボード30側では、コアボード40からの割り込み要求とは関係なしに、ENG制御アプリ31において時間系処理が実行される。この時間系処理は、マザーボード30内のタイマ35により、例えば、1msの周期で実行される。この時間系処理では、ステップ1121においてPCI通信ソフト32によってPCI受信処理が行われる。このとき、非時間系処理で取得された最新の値が使用される。ステップ1122では、ENG制御アプリ31において時間系処理が行われる。そして、ステップ1123においてはPCI送信処理が行われ、データの更新のみが次の非時間系送信処理で送信されてこのルーチンが終了する。

【0059】図12は、図11で説明した通信同期方法の処理の進行状況を時間と共に示したタイミングチャートを示すものである。図11から分かるように、コア

(4) PCI通信方法(フェールセーフ)

PCIバス39を経由したマザーボード30の処理とコアボード40側の処理では、前述のように同期カウンタにより処理の重複を防止するようにハンドリングが実施されるが、同期カウンタが何らかの障害によって故障、或いは破損した場合は、コアボード40側の処理がマザーボード30側の処理の完了待ち状態、或いは、マザー

ボード30側の処理がコアボード40側の処理からの割り込み待ち状態となり、デッドロックループとなることがある。

【0060】そこで、本発明では、このようなデッドロックループ状態を回避するフェールセーフロジックをも

ットロック状態を検出した場合には、強制的に通信を再開するようにした。この処理を図13を用いて説明する。

【0061】図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期(例えば、100μs)になると、ステップ1301において待ちカウンタの値(waitcnt)が判定時間を越えているか否かが判定される。待ちカウンタの値(waitcnt)が、判定時間未満の場合はステップ1303に進み、判定時間を越えている場合はステップ1302において通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)とをクリアしてステップ1303に進む。

【0062】ステップ1303では、通信同期カウンタの値(syncnt)が通信同期カウンタモニタの値(syncnt_m)に等しいか否かが判定される。ステップ1301における待ちカウンタの値(waitcnt)の判定で、待ちカウンタの値(waitcnt)が判定時間未満の場合は、通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)は等しくないで、ステップ1309で待ちカウンタの値(waitcnt)を1だけ更新してこのルーチンを終了する。一方、ステップ1302において通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)とがクリアされた場合は、両者の値は等しいのでステップ1303からステップ1304に進む。

【0063】ステップ1304では、出力情報のセットが行われ、次のステップ1305では入力情報のセットが行われる。続くステップ1306では、待ちカウンタの値(waitcnt)がクリアされ、続くステップ1307においてマザーボード30への割り込み要求が行われる。そして、次のステップ1308で通信同期カウンタモニタの値(syncnt_m)が1だけ更新されてこのルーチンが終了する。マザーボード30側の処理は、図9又は図11で説明した処理と同じであるので、ここではその説明を省略する。

(5) 割り込み制御方法(割り込みイベントのセットとクリア方法)

PCIバス39を経由したマザーボード30とコアボード40の間のデータのやり取りでは、コアボード40で検出した割り込みイベントをPCIバス39を経由してマザーボード30に伝達し、マザーボード30上のENG制御アプリ31の割り込み制御を実現するようにしている。割り込みイベントは、コアボード40からセット、マザーボード30からクリアと、双方向から操作が必要であるが、データの衝突を避けるため、この実施例では、以下のようなPCI通信データを用意し、片方向からの操作で、マザーボード30とコアボード40の間のデータのやり取りを実現するようにしている。

【0064】図14(a)はコアボード40からマザーボード30に送られる各割り込みイベント情報(Dイベント~Aイベント)と、マザーボード30からコアボ

ド40に送られる各割り込みイベントクリア要求（Dクリア要求～Aクリア要求）を示すものである。このような、各割り込みイベント情報と各割り込みイベントクリア要求により、図14（b）に示すような処理が行われる。

【0065】図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期（例えば、100μs）になると、ステップ1401において出力情報のセット処理が行われる。この処理は、割り込みクリア情報を読み込み、セットされているイベント情報をクリアする処理である。出力情報のセット処理が終了すると、ステップ1402において入力情報のセット処理が行われる。この処理は、割り込みイベント情報を疑似マイコン周辺（FPGA）42から取得し、イベント情報をセットする処理である。続くステップ1403においては、マザーボード30への割り込み要求が行われてこのルーチンが終了する。

【0066】マザーボード30側では、このコアボード40からの割り込み要求により、非時間系処理が開始される。ステップ1411では、PCI通信ソフト32によってPCI受信処理が行われる。このPCI受信処理では、割り込みイベントクリア情報[I]をクリアする処理、割り込みイベント情報をPCIバス39経由で読み込む処理、及び、セットされている割り込みイベントに対する割り込みルーチンを起動する処理が行われる。ここで、[I]は内部メモリ31を示し、内部メモリ31にデータを準備しておいて、PCI通信処理部でPCIバス39を経由して共有メモリ46を更新するのである。

【0067】ステップ1411が終了するとステップ1412が実行される。ステップ1412では、非時間系処理（ENG制御アプリ）が実行される。この処理では、各割り込みルーチン（割り込みAルーチン～割り込みDルーチン）が選択されて起動され、起動済みのイベントのクリア情報[I]がセットされる。次のステップ1413ではPCI送信処理が行われる。このPCI送信処理は、割り込みクリア情報をPCIバス39経由で書き込む処理が行われてこのルーチンを終了する。

（6）割り込み制御方法（割り込み優先度管理方法）
これまでのECU1では、割り込みイベントそれぞれに優先度を設定し、各イベントに対する処理の処理優先度を管理していた。そこで、本発明のマイコンのロジック開発装置20では、設定されている割り込み優先度が高い順番で起動処理を行うことにより、実際のECU1に近い割り込み処理を実現するようにしている。例えば、イベント情報の割り込み優先度がBイベント>Aイベント>Dイベント>Cイベント（以後B>A>D>Cのようにイベントは省略）のように設定されている場合には、図15に示すような処理を行う。

【0068】図15のマザーボード30における割り込

み要求処理ルーチンにおけるステップ1501は、図14で説明したステップ1411と同じであり、ステップ1503はステップ1413と同じであるので、ここでは、このルーチンのステップ1502における割り込み優先度の順番で割り込みイベント情報をサーチして、各割り込みルーチンを起動する方法のみについて説明する。

【0069】イベントの割り込み優先度がB>A>D>Cのように設定されている場合、ステップ1502の非時間系処理では、まず、割り込み優先度の高い割り込みBがあるか否かをステップ1511においてサーチして判定する。サーチの結果、割り込みBがない場合はステップ1514に進むが、割り込みBが存在する場合はステップ1512に進み、割り込みBクリア要求をセットする。そして、ステップ1513において割り込みBのルーチンを実行し、このルーチンが終了した後にステップ1514に進む。

【0070】ステップ1514では、次に割り込み優先度の高い割り込みAがあるか否かをサーチして判定する。サーチの結果、割り込みAがない場合はステップ1517に進むが、割り込みAが存在する場合はステップ1515に進み、割り込みAクリア要求をセットする。そして、ステップ1516において割り込みAのルーチンを実行し、このルーチンが終了した後にステップ1517に進む。

【0071】ステップ1517では、3番目に割り込み優先度の高い割り込みDがあるか否かをサーチして判定する。サーチの結果、割り込みDがない場合はステップ1520に進むが、割り込みDが存在する場合はステップ1518に進み、割り込みDクリア要求をセットする。そして、ステップ1519において割り込みDのルーチンを実行し、このルーチンが終了した後にステップ1520に進む。

【0072】ステップ1520では、割り込み優先度が最も低い割り込みCがあるか否かをサーチして判定する。サーチの結果、割り込みCがない場合は割り込みルーチンを終了するが、割り込みCが存在する場合はステップ1521に進み、割り込みCクリア要求をセットする。そして、ステップ1522において割り込みCのルーチンを実行し、このルーチンが終了した後に割り込みルーチンを終了する。

（7）PCI通信処理負荷軽減方法

ENG制御アプリ31の処理は、マザーボード30に採用されるCPUボードの性能向上によって対処することができるが、PCI通信に係る処理は、PCIプロトコルにより制限を受けるため、PCI通信処理負荷は、できるだけ抑えてく必要がある。ここで、割り込みイベント情報が何もない場合、付随する割り込み処理は発生しないため、入力情報をマザーボード30に伝達する必要がなく、また、新たな出力要求も発生しないため、出力

要求をコアボード40に伝達する必要がない。即ち、PCI通信処理は必要がない。

【0073】そこで、本発明では、割り込みイベント情報が何もない場合、PCI通信処理を省略し、PCI通信処理負荷の軽減を図るようにしている。なお、割り込みイベント情報に依存しない時間系処理のために、割り込みイベント情報が何も発生しない状態が継続しても、待ちカウンタの処理により、一定時間毎にはPCI通信が発生することになる。これを図16を用いて説明する。

【0074】図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期（例えば、100μs）になると、ステップ1601において待ちカウンタの値(waitcnt)が判定時間を越えているか否かが判定される。待ちカウンタの値(waitcnt)が、判定時間未満の場合はステップ1603に進み、判定時間を越えている場合はステップ1602において通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)とをクリアしてステップ1603に進む。

【0075】ステップ1603では、通信同期カウンタの値(syncnt)が通信同期カウンタモニタの値(syncnt_m)に等しいか否かが判定される。ステップ1601における待ちカウンタの値(waitcnt)の判定で、待ちカウンタの値(waitcnt)が判定時間未満の場合は、通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)は等しくないので、ステップ1610で待ちカウンタの値(waitcnt)を1だけ更新してこのルーチンを終了する。一方、ステップ1602において通信同期カウンタの値(syncnt)と通信同期カウンタモニタの値(syncnt_m)とがクリアされた場合は、両者の値は等しいのでステップ1603からステップ1604に進む。

【0076】ステップ1604では、出力情報のセットが行われ、次のステップ1605では入力情報のセットが行われる。次のステップ1606では、割り込みイベントが何かあるか否かが判定される。割り込みイベントが何もない場合は、ステップ1610で待ちカウンタの値(waitcnt)を1だけ更新してこのルーチンを終了する。一方、何かしらの割り込みイベントがある場合はステップ1607に進み、このステップ1607では、待ちカウンタの値(waitcnt)がクリアされ、続くステップ1608においてマザーボード30への割り込み要求が行われる。そして、次のステップ1609で通信同期カウンタモニタの値(syncnt_m)が1だけ更新されてこのルーチンが終了する。マザーボード30側の処理は、図9又は図11で説明した処理と同じであるので、ここではその説明を省略する。

(8) タイマ処理方法

ENG制御アプリ31では、コンペア機能を利用したパルス出力要求を算出するため、演算処理内でタイマ値を取得し、パルス出力要求を算出している。従って、タイ

マ値取得要求時には、正確なタイマ値を参照できなければならない。一方、本発明では、タイマ値情報はコアボード40内にあり、マザーボード30側ではPCIバス39を通じて取得する必要がある。ところが、PCI通信間隔は、前述の実施例で説明したように、100μsであり、タイマの係数間隔である1μsに比べて長いため、正確なタイマ値をマザーボード30側で取得するための方策が必要である。

【0077】そこで、本発明では、マザーボード30内の内部タイマ35を使用し、PCI通信で取得したコアボード40からのタイマ値の取得タイミングから、タイマ値取得要求が発生するまでの遅れ時間を補正することによって、マザーボード30側で、正確なタイマ値を取得できるようにしている。この方法について図17を用いて説明する。

【0078】マザーボード30側では、コアボード40からの割り込み要求により、非時間系処理が開始される。ステップ1701では、PCI通信ソフト32によってPCI受信処理が行われる。このPCI受信処理では、タイマ値(T)をコアボード40から取得する処理、及び、内部タイマ値(TO)をラッチする処理が行われる。ステップ1701が終了するとステップ1702が実行される。ステップ1702では、非時間系処理(ENG制御アプリ)が実行される。この処理では、タイマ値取得要求ルーチンが起動される。

【0079】タイマ値取得要求ルーチンでは、まず、ステップ1711において、内部タイマ値(T1)が取得される。次のステップ1712では、タイマ値をコアボード40から取得したタイミング(T)から、現在までの遅れ時間(Td1)を算出する。この算出式は、タイマ値をコアボード40より取得したタイミング(T)を、この時の内部タイマ値(TO)に等しいとして、以下のようになる。

$$【0080】Td1 = T1 - TO$$

このようにして現在までの遅れ時間(Td1)が算出された後は、コアボード40から取得したタイマ値(T)に補正を、以下の式により加えることにより、補正後のタイマ値(T')を算出する。

$$【0081】T' = T + Td1$$

そして、次のステップ1714において、補正後のタイマ値(T')を返してこのルーチンを終了する。

【0082】このようにしてステップ1702が終了した後は、ステップ1703においてPCIの送信処理を行う。

(9) コンペア機能を使用したパルス出力処理方法

ENG制御アプリ31では、例えば、点火出力や燃料噴射出力等のように、コンペア機能を使用してパルス出力要求を行うケースがある。実際のECU1では、コンペア出力端子に一般出力ポート機能とコンペア出力機能の2つの機能があり、これら2つの機能を切り換えてパル

ス出力が行われている。一般出力ポート機能が選択され、即時出力が行われる場合は、この出力はイミディエート出力と呼ばれる。また、コンペア出力機能が選択され、出力時刻、出力レベルが設定されて出力予約が行われる場合は、この出力はタイムド出力と呼ばれる。

【0083】図18(a)はコンペア機能を使用したパルス出力におけるタイムド出力(パターン1)と、イミディエート出力(パターン2)を説明するものである。

【0084】パターン1では、パルス出力開始の所定時間前の時点において、ソフト処理によってタイムド出力を使用してパルス開始出力の予約が行われる。そして、パルス出力開始のコンペア出力開始のコンペアー一致割り込みの中で、タイムド出力を使用したパルス出力終了の予約が行われる。一方、パターン2では、パルス出力開始時点において、イミディエート出力でパルス開始出力が行われると共に、パルス開始出力直後にタイムド出力を使用したパルス終了出力の予約が行われる。

【0085】図18(b)は、マザーボード30における非時間系処理(ステップ1801~ステップ1803)と時間系処理(ステップ1811~ステップ1813)を示すものである。ここで、本構成の処理単位の間(ステップ1801とステップ1803の間と、ステップ1811とステップ1813の間)には、コアボード40に対しては連続した要求を出すことはできない。パターン2のケースでは、イミディエート出力(パルス出力開始)から、タイムド出力(パルス出力終了予約)と連続して処理要求が発生するため、このケースを想定したパルス出力処理が必要となる。

(10) パルス出力要求方法

コンペア機能を使用したあらゆるパルス出力要求パターンに対して、本発明では対応できるようにする。

【0086】図19(a)は、単一処理単位の中でのパルス要求パターンを示すものである。パターン1はタイムド出力であり、コンペア割り込みイベントが必要である。パターン2はイミディエート出力からタイムド出力に移行するものであり、これもコンペア割り込みイベントが必要である。パターン3はイミディエート出力のみである。

【0087】図19(b)はパルス出力要求のデータフォーマットをコンペアの各チャネル毎に示すものである。データフォーマットには、出力予約時刻(タイムド出力用)T_{out}、タイムド出力要求TMREQ、タイムド出力レベルTMLVL、イミディエート出力要求IMMREQ、イミディエート出力レベルIMMLVL、及び、補正の要否ADJがある。

【0088】本発明では、コンペアの各チャネル毎に図20に示すような処理が行われる。図6で説明したコアボード40のタイマ48によって、I/Oサンプリング周期(例えば、100μs)になると、ステップ201において、出力情報のセットの内、パルス出力処理が行

われる。このパルス出力処理では、ステップ2001において、イミディエート出力要求IMMREQがオン状態であるか否かが判定され、オンの場合はステップ2002においてポート出力処理(IMMLVL)が行われてからステップ2003に進む。一方、ステップ2001でイミディエート出力要求IMMREQがオン状態でない場合はステップ2002を行わずにステップ2003に進む。

【0089】ステップ2003では、タイムド要求出力TMREQがオン状態であるか否かが判定され、オン状態の場合はステップ2004においてコンペア出力予約処理(T_{out}/TMLVL)が行われる。一方、ステップ2003でタイムド要求出力TMREQがオン状態でない場合はステップ2004が省略される。次のステップ2002では、入力情報のセットが行われ、その次のステップ2003ではマザーボード30への割り込み要求が行われて、このルーチンが終了する。

【0090】マザーボード30側では、コアボード40からの割り込み要求により、非時間系処理が開始される。ステップ211では、PCI通信ソフト32によってPCI受信処理が行われる。次のステップ212ではENG制御アプリ31内で、パルス出力要求が作成される。パルス出力要求では、ステップ2011においてイミディエート出力要求IMMREQがオールクリアされると共に、タイムド出力要求TMREQもオールクリアされる。次のステップ2012では、イミディエート出力があるか否かが判定され、イミディエート出力がある場合はステップ2013においてイミディエート出力要求IMMREQのセット、イミディエート出力レベルIMMLVLのセットが行われた後にステップ2014に進む。ステップ2012においてイミディエート出力がないと判定された場合は、ステップ2013に進まずにステップ2014に進む。ステップ2014では、タイムド出力があるか否かが判定され、タイムド出力がある場合はステップ2015においてタイムド出力要求TMREQのセット、タイムド出力レベルTMLVLのセット、及び出力予約時刻T_{out}が設定されてステップ212からステップ213に進む。ステップ2014でタイムド出力がないと判定された場合は、ステップ2015を実行せずにステップ2016に進む。ステップ2016においては、PCI送信処理が行われ、ステップ2016においてパルス出力要求をPCIバス39経由で書き込む処理が行われる。

(11) パルス出力補正方法

コンペア機能を使用したパルス出力要求のうち、イミディエート出力でパルス開始、タイムド出力でパルス終了のパルス出力(図18(a)のパターン2)において、実際のECUでの出力パルスと、本発明における出力パルスでは、図21に示すような差Td2が生じる。この差Td2が生じる理由は、実際のECUでは、イミディ

エート出力要求を行うと即時出力されるのに対して、本発明ではイミディエート出力要求によって即時に出力されないからである。即ち、本発明では、図22(a)に示すように、マザーボード30でのイミディエート出力要求がP C Iバス39経由でコアボード40に伝達され、その後のコアボード40のサンプリング処理タイミングで出力されるため、出力要求から実際の出力までに遅れ時間T d 2が発生するのである。

【0091】この結果、実際のE C Uの出力パルスと、本発明の出力パルス共に、タイムド出力要求により、予約時刻で出力されるため、出力タイミングが同一になる。出力パルスの幅にこのような差が生じると、エンジン制御の点火時期は出力パルスのタイミングであるので問題はないが、燃料の噴射量は出力パルスの幅で決まるので、燃料噴射量に差が生じて大きな問題となる。

【0092】この差T d 2を補正するために、イミディエート出力でパルス開始、タイムド出力でパルス終了の要求が出た場合(図20のステップ2012でyes、ステップ2014でもyesとなり、イミディエート出力要求I M M R E Qとタイムド出力要求T M R E Qの両方がセットされている場合)は、次のような処理を行う。

(イ) イミディエート出力要求時、即ち、図22(a)のイミディエート出力要求発生時(A)、の現在時刻をラッチする(T r e q)。

(ロ) 出力実行時、即ち図22(a)のイミディエート出力要求に対する出力を実行時(B)、において、現在時刻(T n o w)とイミディエート出力要求時の時刻

(T r e q)の差(T d 2)を、タイムド出力用の予約時刻に加算する。この時の遅れ時間T d 2は以下の式で表される。

【0093】

$$\text{遅れ時間 } T d 2 = T n o w - T r e q$$

このような処理により、図21に示される実際のE C Uでの出力パルスと、本発明における出力パルスにおける差T d 2は、図22(b)に示すように補正され、パルス幅が同一になる。よって、このパルスはエンジンの燃料噴射に使用することが可能となる。

【0094】ただし、出力信号の種類により、このような補正が必要な場合と、必要でない場合とがある。例えば、エンジンの制御でいえば、点火信号は、パルス出力の終了タイミング(点火時期)が重要であるので、補正は不要である。これに対して、燃料の噴射信号は、パルス出力のパルス幅で噴射量が決まるので、パルス幅が重要であり、前述のような補正が必ず必要である。

【0095】図23は、信号の種類に応じてパルスの終了時点における補正の要、不要を判定した上で補正処理を行うようにした手順を示すものである。

【0096】コアボード40側では、図6で説明したコアボード40のタイマ48によって、I/Oサンプリン

グ周期になると、ステップ231において、出力情報のセットの内、パルス出力処理が行われる。このパルス出力処理では、ステップ2301において、補正が必要か否か、かつ、イミディエート出力要求I M M R E Qがオン状態であるか否か、かつ、タイムド要求出力T M R E Qがオン状態であるか否かが判定され、全てyesの場合にはステップ2302に進んで前述の補正処理が行われる。一方、ステップ2301の判定が全てyesでない場合は補正処理を行わずにステップ2303に進む。

【0097】ステップ2303においては、イミディエート出力要求I M M R E Qがオン状態であるか否かが判定され、オンの場合はステップ2304においてイミディエート出力処理が行われてからステップ2305に進む。一方、ステップ2303でイミディエート出力要求I M M R E Qがオン状態でない場合はステップ2304を行わずにステップ2305に進む。

【0098】ステップ2305では、タイムド要求出力T M R E Qがオン状態であるか否かが判定され、オン状態の場合はステップ2306においてコンベア出力予約処理が行われる。一方、ステップ2305でタイムド要求出力T M R E Qがオン状態でない場合はステップ2306が省略される。次のステップ232では、入力情報のセットが行われ、その次のステップ233ではマザーボード30への割り込み要求が行われて、このルーチンが終了する。

【0099】マザーボード30側では、コアボード40からの割り込み要求により、非時間系処理が開始される。ステップ241では、P C I通信ソフト32によってP C I受信処理が行われる。次のステップ242ではE N G制御アプリ31内で、パルス出力要求が作成される。パルス出力要求では、ステップ2401においてイミディエート出力要求I M M R E Qがオールクリアされると共に、タイムド出力要求T M R E Qもオールクリアされる。次のステップ2402では、イミディエート出力があるか否かが判定され、イミディエート出力がある場合はステップ2403においてイミディエート出力要求I M M R E Qのセット、イミディエート出力レベルI M M L V Lのセット、及び現在時刻(T r e q)のラッチが行われた後にステップ2404に進む。ステップ2402においてイミディエート出力がないと判定された場合は、ステップ2403に進まずにステップ2404に進む。ステップ2404では、タイムド出力があるか否かが判定され、タイムド出力がある場合はステップ2405においてタイムド出力要求T M R E Qのセット、タイムド出力レベルT M L V L Lのセット、及び出力予約時刻T o u tが設定されてステップ242からステップ243に進む。ステップ2404でタイムド出力がないと判定された場合は、ステップ2405を実行せずにステップ243に進む。ステップ213においては、P C I送信処理が行われ、ステップ2406においてパル

ス出力要求をP C Iバス39経由でコアボード40に書き込む処理が行われる。

【0100】なお、以上説明した実施例では、本発明のマイコンのロジック開発装置を用いて開発する電子制御機器として、電子制御式エンジンを説明したが、本発明は、その他の電子制御機器用の組込み用マイコンの開発にも有効に適用できる。更に、本発明のマイコンのロジック開発装置は、次期マイコンの開発に加えて、新規なマイコンの開発にも有効に適用することができる。更に、本発明のマイコンのロジック開発装置は、マザーボードのメモリに記憶するプログラムの変更、コアボードに実装する疑似マイコン周辺を用途に応じて増減することにより、異なるマイコンの開発に対して、繰り返し使用することができる。

【0101】

【発明の効果】以上説明したように、本発明のマイコンのロジック開発装置によれば、ロジックの開発に伴う課題が解消され、C P U機能に対しては新規なロジックや次期ロジックを処理するために必要な処理性能を確保することができ、マイコン周辺リソースに対しては、次期システムに合わせたリソースを確保することができ、新規なロジックや次期ロジックを実現することが可能な組込み用マイコンを短時間で開発することが可能となるという効果がある。また、本発明のマイコンのロジック開発装置は、マイコンの開発に当たって繰り返し利用することが可能であるので、開発コストを低減することができるという効果もある。

【図面の簡単な説明】

【図1】本発明の背景となる電子制御式エンジンの制御システムにおけるE C Uの構成を示すシステム構成図である。

【図2】図1のE C Uのロジックを開発する場合における本発明のマイコンのロジック開発装置の全体構成を示す説明図である。

【図3】本発明のマイコンのロジック開発装置のシステム構成を従来の電子機器制御用のE C Uの構成と比較して示すブロック構成図である。

【図4】図3のマザーボードとコアボードの内部構成の一実施例を示すブロック構成図である。

【図5】(a)は従来のE C Uにおいて入力数が増えた場合の対応を示すブロック構成図、(b)は(a)に対応する本発明のE C Uのブロック構成図である。

【図6】本発明のマイコンのロジック開発装置のマザーボードとコアボードのソフト構成を示す構成図である。

【図7】本発明のマイコンのロジック開発装置における共有メモリを用いたコアボードとマザーボードとの通信方法の一実施例を示すフローチャートである。

【図8】図7の通信方法を実現するためにコアボードに設置される共有メモリの構成及び機能を示す説明図である。

【図9】本発明のマイコンのロジック開発装置におけるマザーボードとコアボードの間のデータ通信の同期方法の一実施例を説明するフローチャートである。

【図10】図9で説明した通信同期方法のタイミングチャートである。

【図11】本発明のマイコンのロジック開発装置におけるマザーボードとコアボードの間のデータ通信の同期方法の別の実施例を説明するフローチャートである。

【図12】図11で説明した通信同期方法のタイミングチャートである。

【図13】本発明のマイコンのロジック開発装置で用いる通信同期方法におけるフェールセーフを説明するフローチャートである。

【図14】(a)は本発明のマイコンのロジック開発装置におけるコアボードで検出した割り込みイベントのセットとクリア方法を説明する図、(b)は本発明のマイコンのロジック開発装置における割り込みイベントの制御方法を示すフローチャートである。

【図15】図14で説明した割り込みイベントの制御方法において割り込みイベントに優先度を設定した場合の割り込み手順を示すフローチャートである。

【図16】本発明のマイコンのロジック開発装置におけるP C Iバスを用いた通信処理における負荷軽減方法を示すフローチャートである。

【図17】本発明のマイコンのロジック開発装置のマザーボードにおけるタイマを用いた処理方法を説明するフローチャートである。

【図18】本発明のマイコンのロジック開発装置におけるコンペア機能を使用したパルス出力処理方法を示すものであり、(a)は2つのパターンの波形図、(b)はマザーボード内の処理手順を示すフローチャートである。

【図19】(a)は本発明のマイコンのロジック開発装置におけるコンペア機能を使用したパルス出力要求パターンの種類を示す組み合わせ図、(b)はパルス出力要求におけるデータフォーマットを示す図である。

【図20】図19のパルス出力要求方法の詳細を示すフローチャートである。

【図21】本発明のマイコンのロジック開発装置におけるコンペア機能を使用したパルス出力要求におけるパルス出力の補正方法を説明する図である。

【図22】(a)は本発明のマイコンのロジック開発装置におけるコンペア機能を使用したパルス出力要求におけるパルス出力の補正方法を説明するタイムチャート、(b)は補正された波形を示す波形図である。

【図23】本発明のマイコンのロジック開発装置におけるコンペア機能を使用したパルス出力要求におけるパルス出力の補正方法の変形例を説明するタイムチャートである。

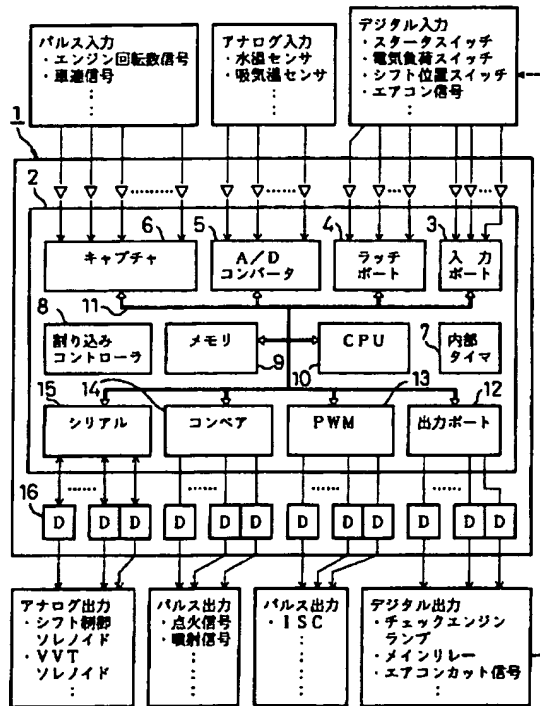
50 【符号の説明】

27

- 1...ECU
- 2...組み込みマイコン
- 7...内部タイマ
- 8...割り込みコントローラ
- 9...メモリ
- 10...CPU
- 11...内部バス
- 20...本発明のロジック開発装置
- 28...ECU入力回路
- 29...ECUコネクタ
- 30...マザーボード
- 31...ソフトウェア
- 32...PCI通信ソフト
- 33...CPU
- 34...仮想割り込みコントローラ
- 35...内部タイマ

【図1】

図1

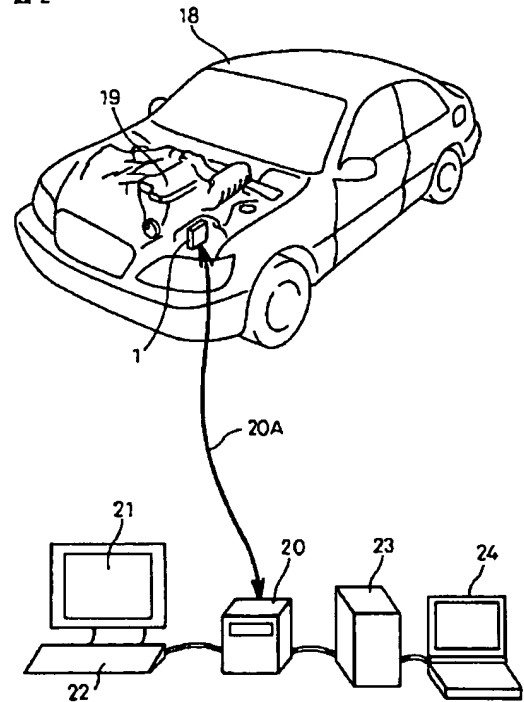


28

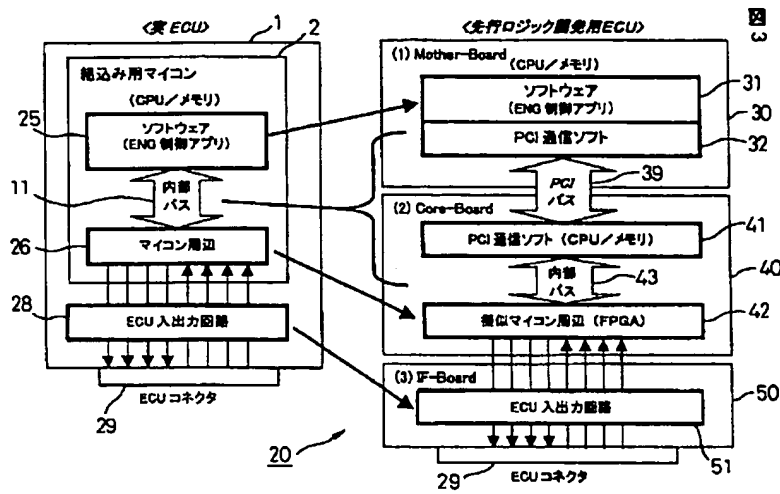
- 36...PCIバスインタフェース
- 37...内部バス
- 38...仮想I/Oレジスタ
- 39...PCUバス
- 40...コアボード
- 41...PCI通信ソフト
- 42...疑似マイコン装置 (FPGA)
- 43...内部バス
- 33...CPU
- 10 44...PCIバスインタフェース
- 45...CPU
- 46...共有メモリ
- 47...内部メモリ
- 48...タイマ
- 50...IFボード

【図2】

図2

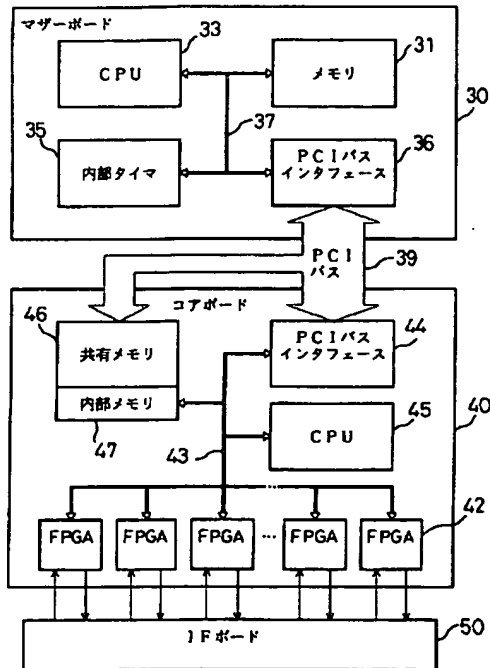


【図3】



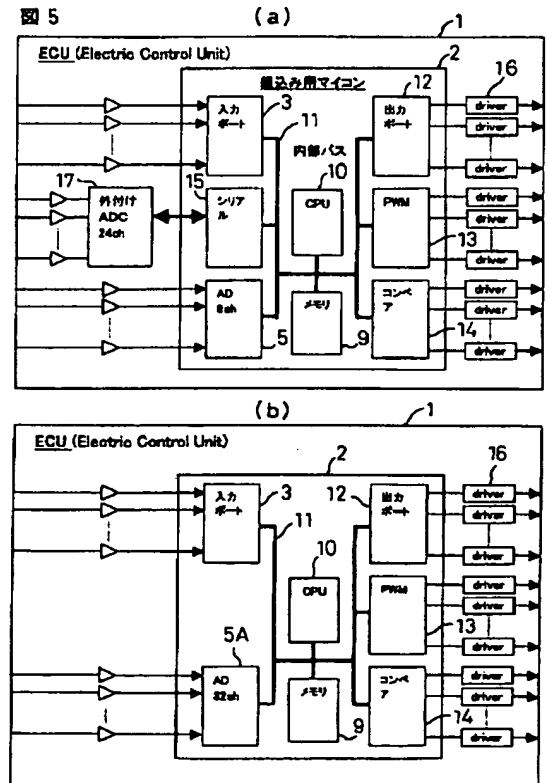
【図4】

図4

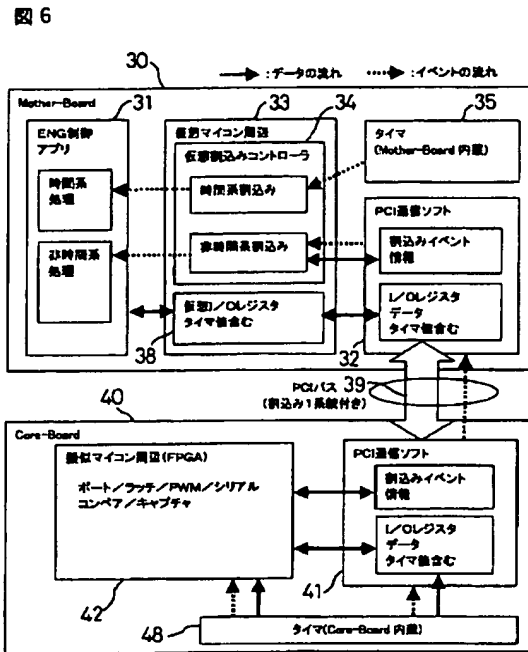


【図5】

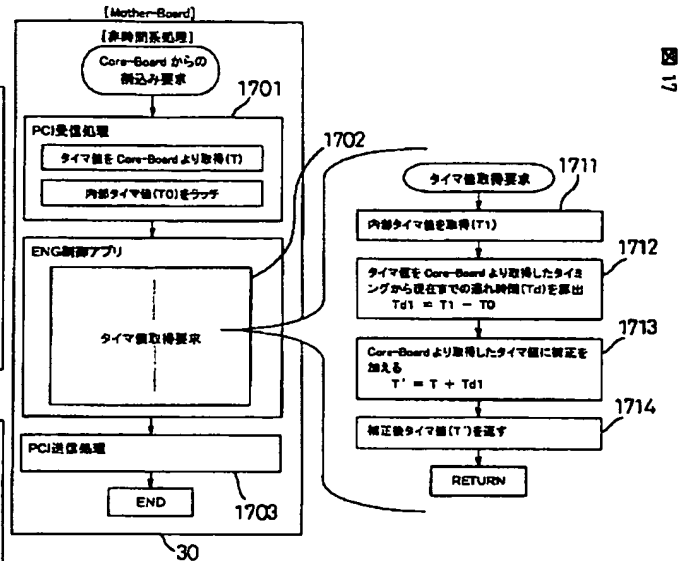
図5



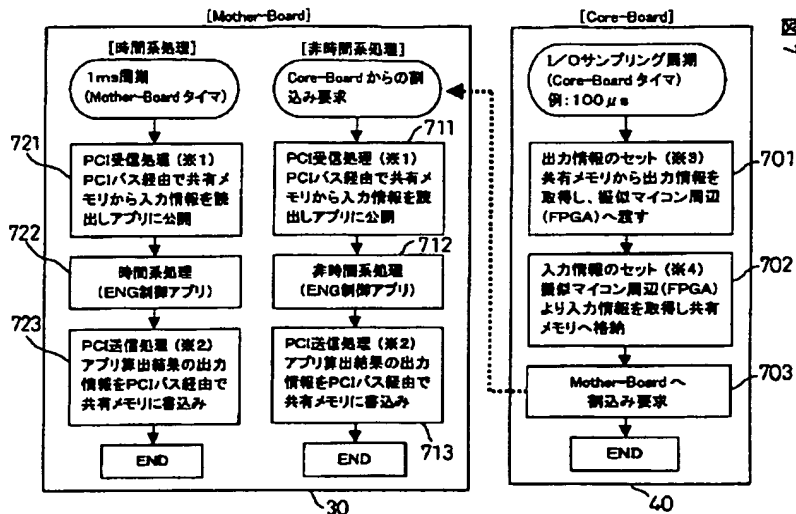
【図6】



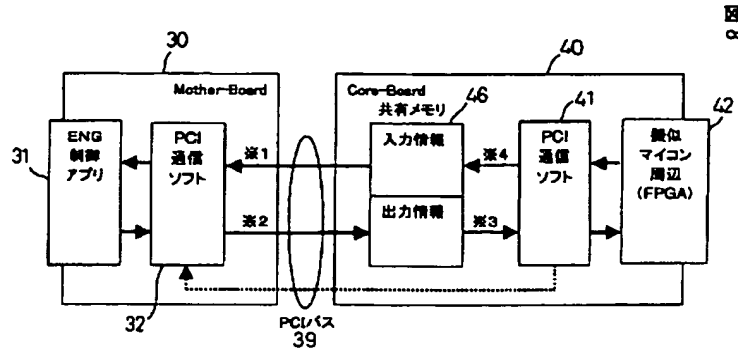
【図17】



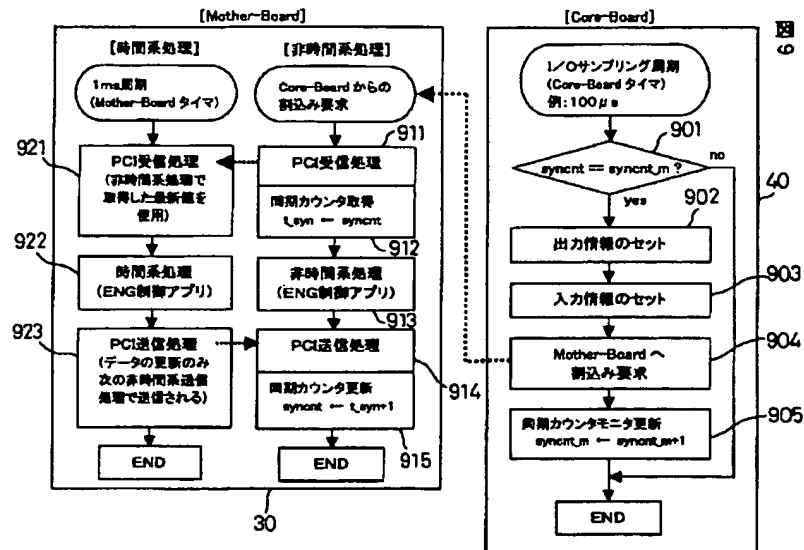
【図7】



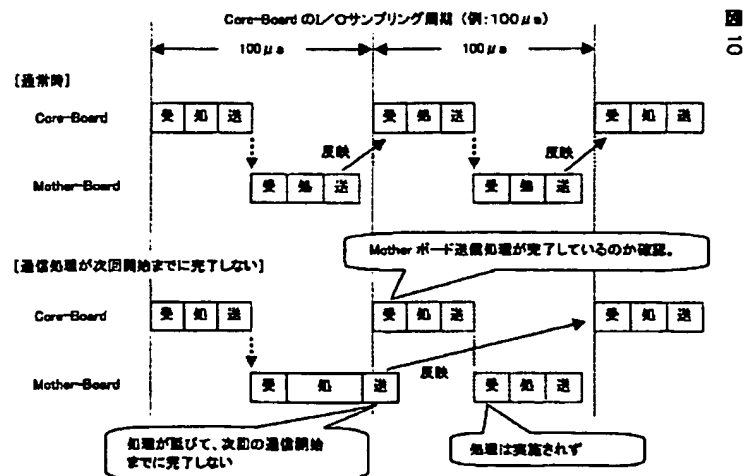
【図8】



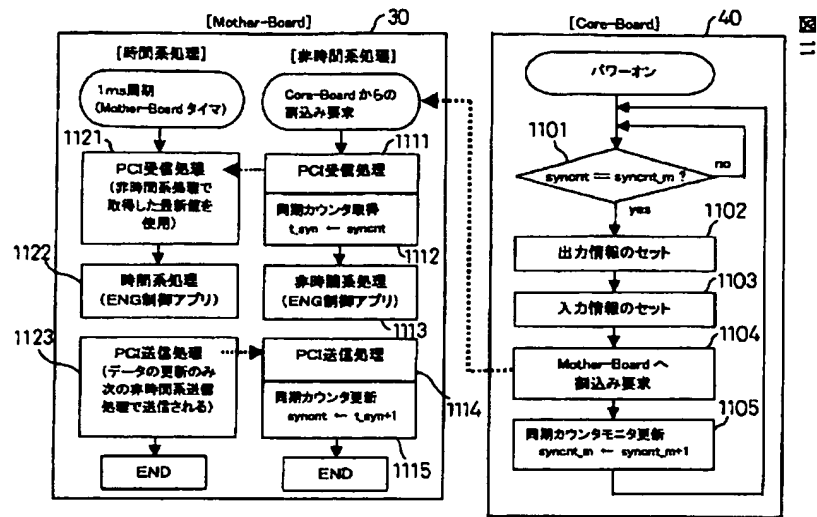
【図9】



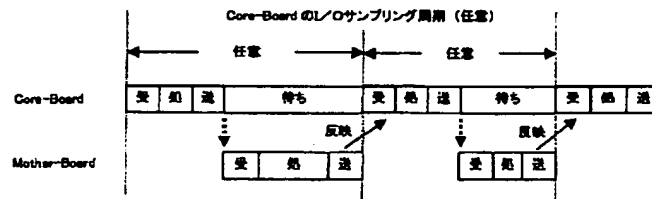
【図10】



【図11】



【図12】



【図19】

【単一処理単位の中でのパルス出力要求パターン】

	内 容	
パターン1	タイムド出力	コンペア割込みイベント要
パターン2	イミディエート出力 → タイムド出力	コンペア割込みイベント要
パターン3	イミディエート出力	—

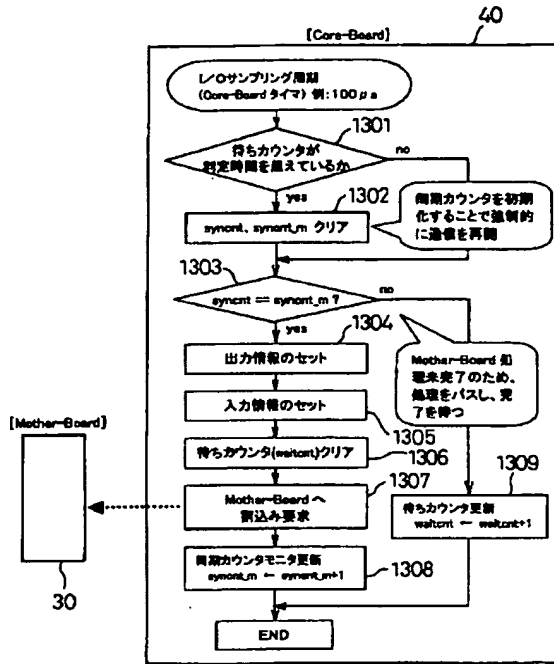
(b)

パルス出力要求データフォーマット (コンペア各ch毎)

出力予約時刻 (タイムド出力用)	タイムド出力要求	タイムド出力 レベル	イミディエート 出力要求	イミディエート 出力レベル	補正要否
Tout	TMREQ	TMLVL	IMMREQ	IMMLVL	ADJ

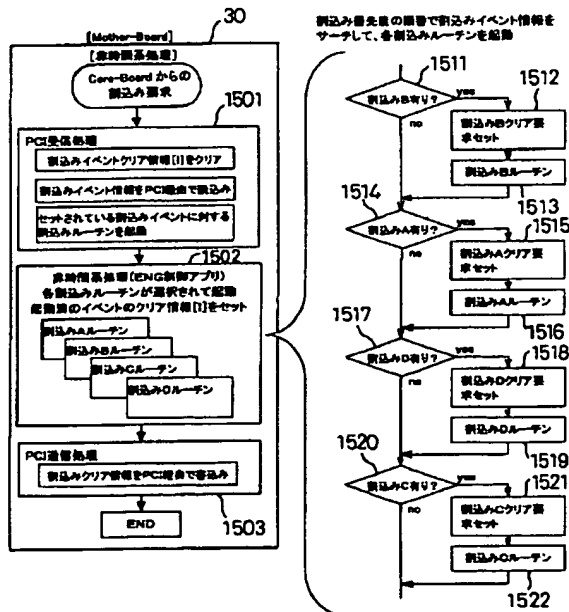
【図13】

図13



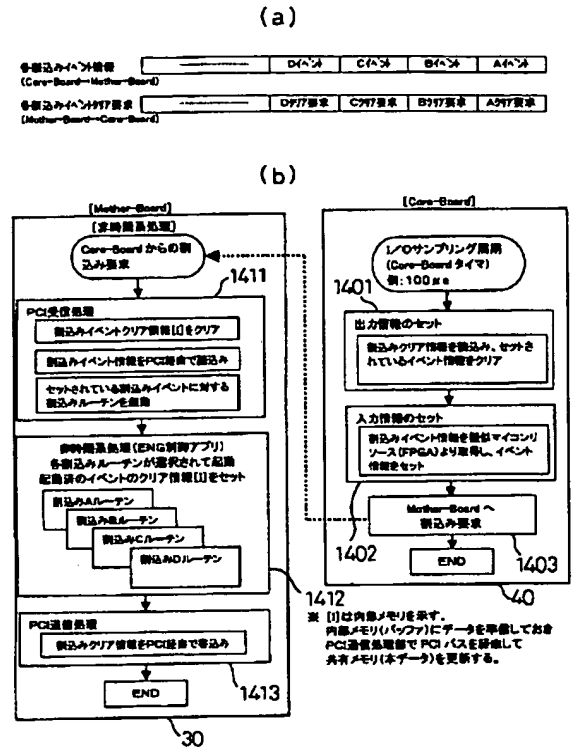
【図15】

図15



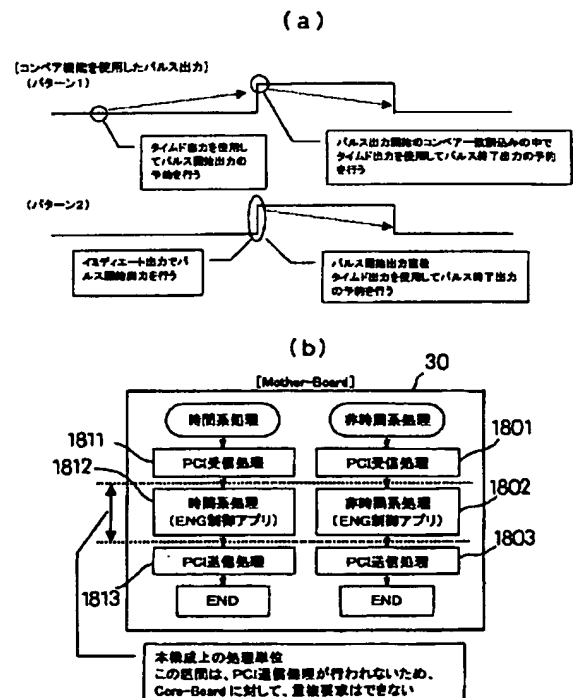
【図14】

図14



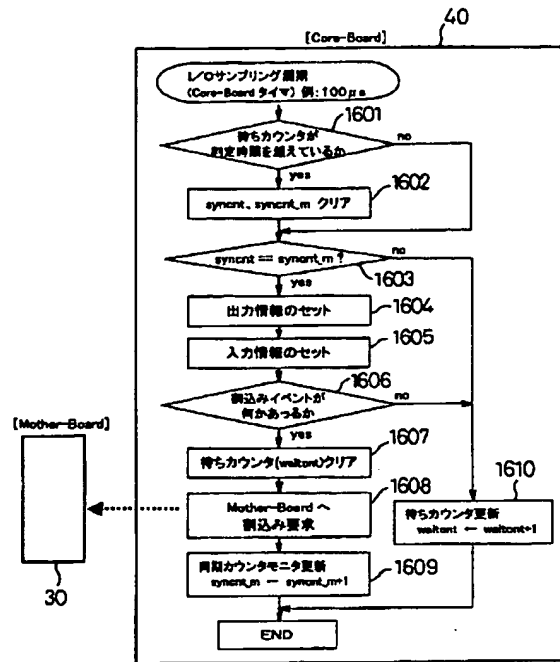
【図18】

図18

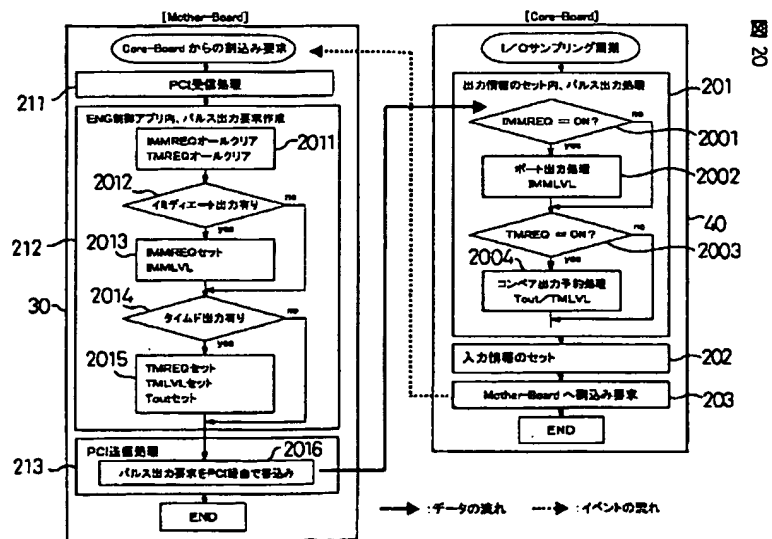


【図16】

図 16

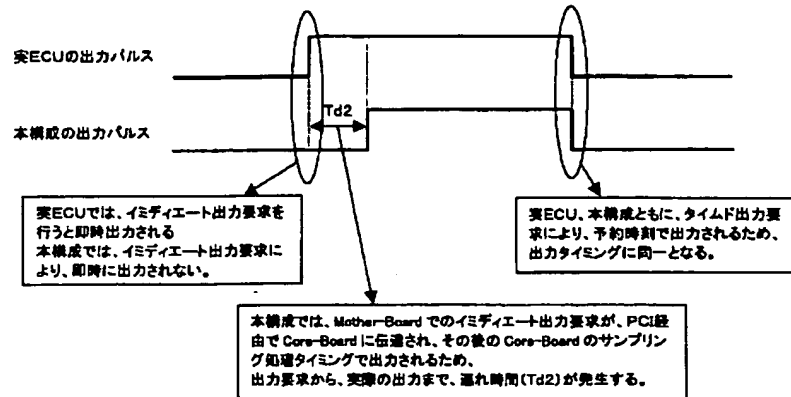


【図20】



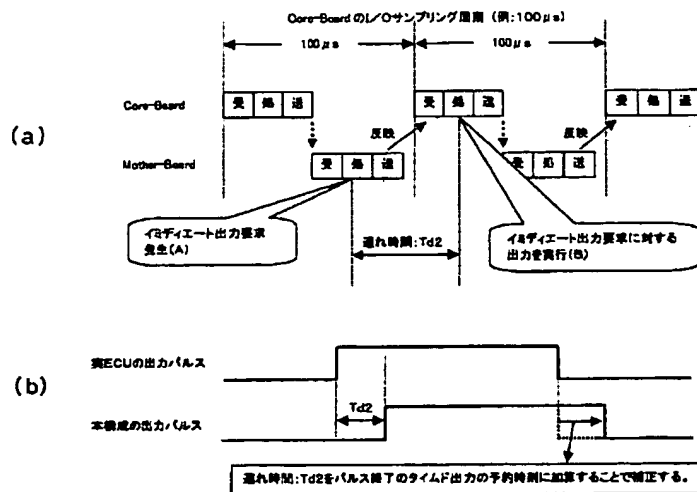
【図21】

図 21

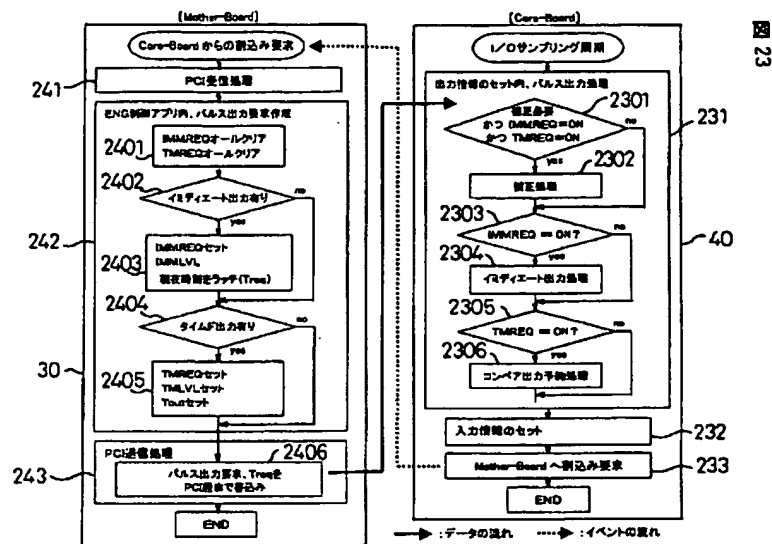


【図22】

図 22



【図23】



フロントページの続き

(72)発明者 柏原 俊浩
兵庫県神戸市兵庫区御所通1丁目2番28号
富士通テン株式会社内

Fターム(参考) 5B042 G008 HH01 HH06
5B048 AA14 DD17 FF03

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.